

Convolución adaptativa, rápida y con poca memoria para ecuaciones de evolución con memoria

M. LÓPEZ-FERNÁNDEZ¹, C. LUBICH², A. SCHÄDLE³

¹ *Dpto. de Matemáticas, Universidad Autónoma de Madrid, Campus de Cantoblanco, Ctra. de Colmenar Viejo Km. 15, E-28049 Madrid. E-mail: maria.lopez@uam.es.*

² *Mathematisches Institut, Universität Tübingen, Auf der Morgenstelle 10, D-72076 Tübingen, Germany. E-mail: lubich@na.uni-tuebingen.de.*

³ *ZIB Berlin, Takustr. 7, D-14195 Berlin, Germany. E-mail: schaedle@zib.de.*

Palabras clave: convolution quadrature, adaptivity, Volterra integral equations, numerical inverse Laplace transform, anomalous diffusion.

Resumen

Presentamos un algoritmo de paso variable para aproximar convoluciones que aparecen como términos con memoria en ecuaciones de evolución. Para avanzar N pasos, el algoritmo requiere sólo $O(N \log N)$ operaciones y $O(\log N)$ datos en la memoria activa, en lugar de las $O(N^2)$ operaciones y $O(N)$ datos en la memoria necesarios para una implementación más directa. Una característica básica del algoritmo es la reducción, vía integrales de contorno, a ecuaciones diferenciales escalares, que se integran numéricamente con paso variable. En lugar del núcleo de convolución, el método utiliza los valores de su transformada de Laplace. Ilustramos el algoritmo aplicándolo a un modelo para reacciones químicas con difusión inhibida.

1. Introducción

Consideramos el cálculo de convoluciones

$$\int_0^t f(t - \tau) g(\tau) d\tau, \quad 0 \leq t \leq T, \quad (1)$$

posiblemente con núcleo matricial f y función vectorial g , y la resolución de ecuaciones de evolución con memoria conteniendo integrales de convolución de este tipo donde g no es una función conocida a priori, sino que $g(\tau)$ depende de la solución en tiempo τ de la ecuación integral o integro-diferencial.

En trabajos anteriores [7, 3, 8], se han desarrollado algoritmos que son rápidos y con poca memoria: para aproximar (1) en una secuencia $t = nh$ ($n = 0, 1, \dots, N$) con tamaño de paso constante h y $Nh = T$, estos algoritmos necesitan $O(N \log N)$ operaciones, $O(\log N)$ evaluaciones de la transformada de Laplace $F = \mathcal{L}f$, ninguna de f , y $O(\log N)$ entradas en la memoria activa. En el paso n -ésimo, g se evalúa en $t_n = nh$, pero el algoritmo “olvida” la historia $g(t_j)$ para $j < n$ y sólo guarda en memoria unas pocas combinaciones lineales de dichos valores de g . Esto contrasta con las $O(N^2)$ operaciones, $O(N)$ evaluaciones del núcleo f , y $O(N)$ datos en memoria necesarios para una implementación más directa de una fórmula de cuadratura para (5). Señalamos además que en muchas aplicaciones es la transformada de Laplace F la que se conoce de antemano, en lugar del núcleo f .

Una característica básica del algoritmo rápido que presentamos aquí es la reducción, vía la representación mediante integrales de contorno, a ecuaciones diferenciales de la forma $y' = \lambda y + g$ para ciertos $\lambda \in \mathbb{C}$, que son aproximadas numéricamente. La integración de estas ecuaciones diferenciales no tiene que hacerse necesariamente con paso temporal fijo h , como se hace en [3, 7, 8], sino que el tamaño del paso se puede adaptar al comportamiento de g . Esta observación es el punto de partida del algoritmo “adaptativo”, rápido y con poca memoria que presentamos aquí. La necesidad de utilizar paso de tiempo variable en las aplicaciones para resolver ecuaciones de evolución integro-diferenciales aparece en diversas referencias de la literatura [1, 2]. Sin embargo, ninguno de los algoritmos propuestos en estas referencias aprovecha la estructura de convolución para reducir las $O(N^2)$ operaciones y $O(N)$ datos en la memoria necesarios para dar N pasos de tiempo.

El algoritmo de convolución propuesto aquí funciona en caso de que la transformada de Laplace F sea *sectorial*:

$$\begin{aligned} &F \text{ es analítica en un sector } |\arg(s - \sigma)| < \pi - \varphi \text{ con } \varphi < \frac{1}{2}\pi, \\ &\text{y en este sector, } |F(s)| \leq M |s|^{-\nu} \text{ para algún } M \text{ real y } \nu > 0, \end{aligned} \quad (2)$$

siendo un ingrediente esencial la inversión numérica de la transformada F . Para ello, discretizamos la fórmula de inversión

$$f(t) = \frac{1}{2\pi i} \int_{\Gamma} e^{t\lambda} F(\lambda) d\lambda, \quad t > 0, \quad (3)$$

tomando como Γ un contorno apropiado en el sector de holomorfía de F . Con un sólo contorno Γ no podemos obtener una aproximación uniformemente buena para todo $t \in (0, T]$, así que utilizamos diferentes hipérbolas Γ_ℓ para intervalos de aproximación uniforme que podemos tomar con crecimiento geométrico, del tipo $[B^{\ell-1}h_*, B^{\ell+1}h_*]$ con $B \in \mathbb{N}$, $B \geq 1$, digamos $B = 5$, y con un paso mínimo h_* . El número necesario de contornos está por tanto acotado por $\log_B(T/h_*)$, en lugar del $\log_B(N)$ para el algoritmo de paso fijo. Para más ejemplos y detalles véase [4].

2. Preparación: La inversión numérica de la transformada de Laplace

En la fórmula de inversión (3), elegimos Γ como la rama izquierda de una hipérbola parametrizada por $\gamma(x) = \mu(1 - \sin(a + ix)) + \sigma$, donde $\mu > 0$ es un parámetro de escala,

σ es el desplazamiento en (2) y $0 < a < \pi/2 - \varphi$. De este modo Γ es la rama izquierda de la hipérbola con centro en $(\mu, 0)$, focos en $(0, 0)$, $(2\mu, 0)$ y asíntotas formando ángulos $\pm(\pi/2 + a)$ con el eje real, de modo que Γ permanece dentro del sector (2) de holomorfia de F . Tras parametrizar (3), aproximamos f aplicando a la integral a lo largo del eje real que resulta la regla del trapecio truncada, esto es

$$f(t) = \frac{1}{2\pi i} \int_{\Gamma} e^{t\lambda} F(\lambda) d\lambda \approx \sum_{k=-K}^K w_k e^{t\lambda_k} F(\lambda_k), \quad (4)$$

donde los pesos w_k y los nodos de cuadratura λ_k están dados por $w_k = \frac{\tau}{2\pi i} \gamma'(k\tau)$ y $\lambda_k = \gamma(k\tau)$, con $\tau > 0$ un parámetro apropiado. El buen comportamiento de esta fórmula de cuadratura para aproximar (3) se debe a las buenas propiedades de la regla del trapecio cuando el integrando se puede prolongar analíticamente a una banda horizontal en torno al eje real [9, 10]. Enunciamos a continuación la siguiente cota de error, que decae exponencialmente con respecto al número de nodos de cuadratura. Para más detalles, nos referimos a [5] y [6].

Teorema 1 [6] *Supongamos que la transformada de Laplace F satisface la condición de sectorialidad (2). Para $T > 0$, $\Lambda \geq 1$, $0 < a < \pi/2 - \varphi$, y $K \geq 1$ fijos, existen cantidades positivas C_1, C_2, C, c que dependen de a y Λ (C también depende de T a menos que $\sigma < 0$ en (2)) tales que la selección de parámetros $\tau = C_1/K$ y $\mu = C_2K/(\Lambda t_0)$ da lugar a un error de cuadratura en (4) acotado por*

$$|E_K(t)| \leq C t^{\nu-1} (\epsilon + e^{-cK}),$$

uniformemente para $t \in [t_0, \Lambda t_0]$, con $0 < t_0 \leq T/\Lambda$ arbitrario. En la cota, ν es el exponente en (2) y ϵ la precisión en las evaluaciones de la transformada de Laplace F y en las operaciones elementales involucradas en (4).

De este modo, $K = O(\log \frac{1}{\epsilon})$ nodos de cuadratura son suficientes para obtener una precisión $O(\epsilon)$ en la aproximación de la integral de contorno (3), uniformemente para $t \in [t_0, \Lambda t_0]$. En la práctica, tomamos $a \approx \frac{1}{2}(\frac{\pi}{2} - \varphi)$ y calculamos los valores de C_1 y C_2 siguiendo el proceso de optimización descrito en [6].

3. El algoritmo de convolución de paso variable, rápido y con poca memoria

3.1. Reducción local a ecuaciones diferenciales

El objetivo es aproximar

$$u(t) = \int_0^t f(t - \tau) g(\tau) d\tau \quad (5)$$

en una secuencia de tiempos $0 < t_1 < \dots < t_N$, espaciados arbitrariamente. Para un t_n dado, insertamos la fórmula de inversión de la transformada de Laplace en (5) y escribimos

$$\int_0^{t_n} f(t_n - \tau) g(\tau) d\tau = \int_0^{t_n} \frac{1}{2\pi i} \int_{\Gamma} e^{(t_n - \tau)\lambda} F(\lambda) d\lambda g(\tau) d\tau. \quad (6)$$

La inversión numérica de la transformada de Laplace se lleva a cabo aplicando la cuadratura (4). Como hemos visto en la Sección 2, se puede utilizar el mismo contorno Γ para aproximar f en diferentes valores de $t \in [t_0, \Lambda t_0]$, para un ratio dado $\Lambda \geq 1$. En (6) necesitamos aproximar $f(t_n - \tau)$ para $t_n - \tau \in [0, t_n]$, de modo que no podemos utilizar un único contorno Γ y necesitamos dividir la integral en (5) en varias partes. Para tiempos intermedios apropiados $0 < t^- < t^+ < t_n$, con $(t_n - t^-)/(t_n - t^+) \leq \Lambda$, seleccionamos un contorno adecuado Γ para cada intervalo $[t_n - t^+, t_n - t^-]$ y aproximamos

$$\begin{aligned} \int_{t^-}^{t^+} f(t_n - \tau)g(\tau) d\tau &= \int_{t^-}^{t^+} \frac{1}{2\pi i} \int_{\Gamma} e^{(t_n - \tau)\lambda} F(\lambda) d\lambda g(\tau) d\tau \\ &\approx \int_{t^-}^{t^+} \sum_{k=-K}^K w_k e^{(t_n - \tau)\lambda_k} F(\lambda_k) g(\tau) d\tau = \sum_{k=-K}^K w_k F(\lambda_k) e^{(t_n - t^+)\lambda_k} y(t^+, t^-, \lambda_k), \end{aligned}$$

donde $y(t^+, t^-, \lambda_k)$ es la solución en t^+ de la EDO lineal no homogénea $y' = \lambda_k y + g$, con $y(t^-) = 0$, para $-K \leq k \leq K$. Aproximamos $y(t^+, t^-, \lambda_k)$ interpolando g linealmente a trozos en los t_j , $j = 0, \dots, N-1$, e integrando de forma exacta. Denotando por $\bar{y}(t^+, t^-, \lambda_k)$ a la aproximación resultante, calculamos:

$$\int_{t^-}^{t^+} f(t_n - \tau) g(\tau) d\tau \approx \sum_{k=-K}^K w_k F(\lambda_k) e^{(t_n - t^+)\lambda_k} \bar{y}(t^+, t^-, \lambda_k). \quad (7)$$

3.2. Llenado del mosaico

La clave del algoritmo es la forma que en que se seleccionan los tiempos para dividir la integral en (5). Para ello, seguimos el esquema representado por el mosaico en el triángulo $\{(t, \tau) : 0 \leq \tau \leq t \leq T\}$ que mostramos en la Figura 1, donde las piezas crecen geométricamente a medida que aumenta la distancia a la diagonal. Por el momento fijamos un paso mínimo h_* . Si en el interior del triángulo la línea vertical de $(t_n, 0)$ a (t_n, t_n) corta a L piezas distintas del mosaico, obtenemos los valores t_ℓ^- y t_ℓ^+ , para $1 \leq \ell \leq L$, como el menor y el mayor de los tiempos t_j , $1 \leq j \leq n$, contenidos en la pieza ℓ -ésima respectivamente. Cuando la pieza ℓ -ésima contiene alguno de los tiempos del mallado, el índice ℓ se recoge en un conjunto de índices J . Los valores $t_\ell^\pm = t_{n,\ell}^\pm$ dependen de n aunque por sencillez omitimos n en la notación. Las piezas del mismo tamaño forman una clase que representa la distancia a la diagonal en el mosaico y corresponden por tanto a un intervalo de aproximación distinto y un contorno diferente para realizar la inversión de la transformada, lo que se traduce en un conjunto distinto de $2K + 1$ ecuaciones diferenciales escalares. Los intervalos de aproximación para los valores $t_n - t_\ell^\pm$ son del tipo $I_\ell = [B^{\ell-1}h_*, B^{\ell+1}h_*]$, $2 \leq \ell \leq L$, de modo que el ratio Λ es B^2 . Dado que los tiempos t_j no están equiespaciados, en esta partición aparecen “huecos” entre los valores $t_{\ell+m}^+$ y t_ℓ^- , que corresponden en la Figura 1 a pares de líneas horizontales con alguna frontera de las piezas del mosaico enmedio. De este modo, dividimos (5) en $2|J| + 1$ partes

$$u(t_n) = \tilde{u}^{(0)}(t_n) + \sum_{\ell \in J} u^{(\ell)}(t_n) + \sum_{\ell \in J} \tilde{u}^{(\ell)}(t_n) \quad (8)$$

donde

$$u^{(\ell)}(t_n) = \int_{t_\ell^-}^{t_\ell^+} f(t_n - \tau)g(\tau) d\tau \quad (9)$$

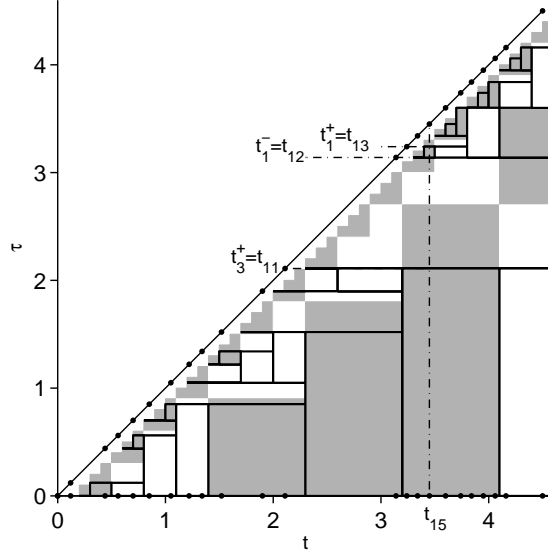


Figura 1: Mosaico en el triángulo $\tau \leq t$ para $B = 3$ con los tiempos t_j indicados por puntos. Cada rectángulo monocromático completamente cerrado por líneas negras corresponde a un conjunto de soluciones $y(t_\ell^+, t_\ell^-, \lambda_k)$, $1 \leq k \leq 2K + 1$, de EDOs lineales.

se aproxima siguiendo (7) y

$$\tilde{u}^{(0)}(t_n) = \int_{t_{n-1}}^{t_n} f(t_n - \tau)g(\tau) d\tau \quad \text{y} \quad \tilde{u}^{(\ell)}(t_n) = \int_{t_{\ell+m}^+}^{t_\ell^-} f(t_n - \tau)g(\tau) d\tau \quad (10)$$

corresponden al paso de t_{n-1} a t_n y a los “huecos” entre $t_{\ell+m}^+$ y t_ℓ^- , respectivamente (véase la Figura 1). Estas partes se calculan mediante “pasos directos” que explicamos en la siguiente sección. El procedimiento formal es el siguiente: dado un paso mínimo h_* y una base B , para cada t_n tomamos L el menor entero tal que podemos representar $\lceil t_n/h_* \rceil = 2 + \sum_{\ell=1}^L b_\ell B^{\ell-1}$ con $b_\ell \in \{1, 2, \dots, B\}$, donde $\lceil x \rceil$ denota el menor de los enteros mayores que x . t_ℓ^+ es el mayor y t_ℓ^- el menor de los valores en $\{t_j : j = 0, \dots, n\}$, tales que $h_* \sum_{k=\ell+1}^L b_k B^{k-1} \leq t_\ell^- \leq t_\ell^+ \leq h_* \sum_{k=\ell}^L b_k B^{k-1}$. Los t_ℓ^\pm son tales que los límites de integración en $u^{(\ell)}(t_n)$ verifican $t_n - t_\ell^\pm \in I_\ell$. El llenado del mosaico representado en la Figura 1 se lleva a cabo avanzando todas las ecuaciones diferenciales para todos los contornos en cada paso de tiempo, de modo que el algoritmo va “olvidando” los valores pasados de g , a excepción los valores en t_ℓ^\pm , necesarios para los pasos directos descritos a continuación. En total se guardan tres copias de las soluciones a las EDOs. Pseudocódigos para la organización de esta descomposición y más detalles se pueden encontrar en [4].

3.3. Pasos directos

Los valores $\tilde{u}^{(\ell)}(t_n)$ que cubren los “huecos” $[t_{\ell+m}^+, t_\ell^-]$ están dados por

$$\tilde{u}^{(\ell)}(t_n) = \int_{t_{\ell+m}^+}^{t_\ell^-} f(t_n - \tau)g(\tau) d\tau = \int_{t_j}^{t_{j+1}} f(t_n - \tau)g(\tau) d\tau, \quad (11)$$

para algún $j \in \{0, 1, \dots, N-1\}$. En $[t_j, t_{j+1}]$, aproximamos $g(t)$ por la función lineal

$$g(t) \approx \bar{g}(t) = g_j + \frac{g_{j+1} - g_j}{h_{j+1}}(t - t_j), \quad h_{j+1} = t_{j+1} - t_j,$$

con $g_j = g(t_j)$, $j = 0, 1, \dots, N$. Prolongando $\bar{g}(t)$ a todo $[0, t_n]$, calculamos

$$\begin{aligned} \int_{t_j}^{t_{j+1}} f(t_n - \tau) \bar{g}(\tau) d\tau &= \int_{t_j}^{t_n} f(t_n - \tau) \bar{g}(\tau) d\tau - \int_{t_{j+1}}^{t_n} f(t_n - \tau) \bar{g}(\tau) d\tau \\ &= \mathcal{L}^{-1}[F \cdot \mathcal{L}\bar{g}(\cdot + t_j)](t_n - t_j) - \mathcal{L}^{-1}[F \cdot \mathcal{L}\bar{g}(\cdot + t_{j+1})](t_n - t_{j+1}) \\ &= \mathcal{L}^{-1} \left[F_1 g_j + F_2 \frac{g_{j+1} - g_j}{h_{j+1}} \right] (t_n - t_j) - \mathcal{L}^{-1} \left[F_1 g_{j+1} + F_2 \frac{g_{j+1} - g_j}{h_{j+1}} \right] (t_n - t_{j+1}), \end{aligned}$$

donde $F_1(s) = F(s)/s$ y $F_2(s) = F(s)/s^2$. Aproximamos las transformadas inversas $f_1(t) = (\mathcal{L}^{-1}F_1)(t)$ y $f_2(t) = (\mathcal{L}^{-1}F_2)(t)$ en $t = t_n - t_j$ y $t = t_n - t_{j+1}$ utilizando de nuevo (4) con los contornos correspondientes a los intervalos de aproximación I_{ℓ_1} y I_{ℓ_2} con $t_n - t_{j+1} \in I_{\ell_2}$ y $t_n - t_j \in I_{\ell_1}$ (notamos que $f_1(0) = f_2(0) = 0$). El paso directo se aproxima entonces por la combinación lineal

$$\begin{aligned} \int_{t_j}^{t_{j+1}} f(t_n - \tau) \bar{g}(\tau) d\tau &= f_1(t_n - t_j) g_j + f_2(t_n - t_j) \frac{g_{j+1} - g_j}{h_{j+1}} \\ &\quad - f_1(t_n - t_{j+1}) g_{j+1} - f_2(t_n - t_{j+1}) \frac{g_{j+1} - g_j}{h_{j+1}}. \end{aligned} \tag{12}$$

Nota acerca de la complejidad. Dada una secuencia de tiempos cualquiera $0 < t_1 < \dots < t_N = T$ con paso mínimo $h_* = \min_j(t_{j+1} - t_j)$, el algoritmo aproxima (5) con error menor que ε realizando un total de $O(NLK) = O(N \log \frac{T}{h_*} \log \frac{1}{\varepsilon})$ operaciones y guardando en memoria $O(LK) = O(\log \frac{T}{h_*} \log \frac{1}{\varepsilon})$ vectores del tamaño de g .

Notemos que el algoritmo es independiente del método elegido para seleccionar el tamaño de los pasos en la integración numérica de las ecuaciones integrales e integro-diferenciales. En el ejemplo siguiente la estrategia seguida se basa en controlar el error en la interpolación lineal a trozos de g en (5), que por otra parte es la única fuente de error en el esquema aparte de la discretización de las integrales de contorno, pero otras alternativas son posibles. Para más ejemplos véase [4].

4. Dinámica de reacciones químicas con difusión inhibida

Consideramos tres especies moleculares A , B y C , con concentraciones u_1 , u_2 y u_3 respectivamente, que reaccionan en la forma $A + B \xrightarrow{k_1} C$, $C \xrightarrow{k_2} A + B$, $C \xrightarrow{k_3} A + P$, donde P es el producto resultante. La difusión de las especies es anómala, dando lugar a una ecuación de reacción difusión con un término de memoria. Siguiendo el modelo propuesto en [11], (cf. [2]), consideramos el siguiente sistema de ecuaciones integro-diferenciales:

$$\begin{aligned} u_1(t) - u_1(0) &= \partial_t^{-\alpha} [K \Delta u_1(t) - k_1 u_1(t) u_2(t) + (k_2 + k_3) u_3(t)] \\ u_2(t) - u_2(0) &= \partial_t^{-\alpha} [K \Delta u_2(t) - k_1 u_1(t) u_2(t) + k_2 u_3(t)] \\ u_3(t) - u_3(0) &= \partial_t^{-\alpha} [K \Delta u_3(t) + k_1 u_1(t) u_2(t) - (k_2 + k_3) u_3(t)], \end{aligned} \tag{13}$$

donde $\Delta = \partial_{xx}$ es el Laplaciano en 1D con condiciones de contorno periódicas en $[-5, 5]$ y $\partial_t^{-\alpha}$ denota la integral fraccionaria de orden $0 < \alpha < 1$, definida por el operador de Riemann Liouville

$$\partial_t^{-\alpha} g(t) = \frac{1}{\Gamma(\alpha)} \int_0^t (t - \tau)^{\alpha-1} g(\tau) d\tau. \quad (14)$$

El núcleo de convolución $f(t) = t^{\alpha-1}/\Gamma(\alpha)$ es débilmente singular y tiene transformada de Laplace $F(s) = s^{-\alpha}$. Elegimos $K = 0.5$, $k_1 = 1$, $k_2 = 2$ y $k_3 = 3$, e integramos hasta $T = 30$. Fijamos $\alpha = 0.5$ y consideramos funciones escalón suavizadas como dato inicial. Denotando $\mathbf{u} = [u_1, u_2, u_3]^T$, I_3 la matriz identidad 3×3 , y siguiendo la notación de la Sección 3.3 para los pasos directos, obtenemos la ecuación discreta

$$\begin{aligned} \left(I_3 \otimes I - \frac{f_2(h_n)}{h_n} (K I_3 \otimes S + R) \right) \mathbf{u}^n &= \mathbf{u}_0 + \frac{1}{\Gamma(\alpha)} \int_0^{t_{n-1}} \frac{g(\bar{\mathbf{u}}(\tau))}{(t - \tau)^{1-\alpha}} d\tau \\ &+ \left(f_1(h_n) - \frac{f_2(h_n)}{h_n} \right) (K I_3 \otimes S + R) \mathbf{u}^{n-1} + k_1 f_1(h_n) \mathbf{e} \otimes u_1^{n-1} u_2^{n-1}, \end{aligned}$$

donde $\bar{\mathbf{u}}$ denota el interpolante lineal a trozos de \mathbf{u} en los tiempos t_0, t_1, \dots, t_n y, para M nodos en la discretización espacial y \mathbf{v} un vector columna de longitud $3M$, definimos

$$g(\mathbf{v}) = (K I_3 \otimes S + R) \mathbf{v} + k_1 (\mathbf{e} \otimes v_1 v_2), \quad \mathbf{e} = [-1, -1, 1]^T,$$

con S la aproximación de ∂_{xx} por diferencias finitas de segundo orden y condiciones de contorno periódicas y R la matriz $3M \times 3M$

$$R = \begin{pmatrix} 0 & 0 & (k_2 + k_3)I_M \\ 0 & 0 & k_2 I_M \\ 0 & 0 & -(k_2 + k_3)I_M \end{pmatrix},$$

con I_M la matriz identidad $M \times M$.

Dada una tolerancia Tol , la estrategia para elegir el nuevo paso h_{n+1} es

$$Ch_{n+1}^2 \gamma'_n = 0.8 \cdot Tol, \quad (15)$$

con $\gamma'_n = \|\bar{g}'(t_n)\|$ y $C \approx \int_0^T |f(t)| dt$. Adicionalmente imponemos $h_n/4 < h_{n+1} < 4h_n$. El test para el nuevo paso propuesto h_{n+1} es $Ch_{n+1}^2 \gamma'_{n+1} \leq Tol$, donde el nuevo valor $g(t_n + h_{n+1})$ se utiliza en el cálculo de γ'_{n+1} . Si se cumple el test, aceptamos h_{n+1} y tomamos $t_{n+1} = t_n + h_{n+1}$. Si no, repetimos el procedimiento con h_{n+1} obtenido a partir de (15) con γ'_{n+1} en lugar de γ'_n . Este procedimiento se repite si es necesario hasta que se verifique el test.

La Figura 2 muestra los resultados numéricos obtenidos al aplicar el algoritmo. La inversión de la transformada de Laplace se ha llevado a cabo con $a = 1$, $d = 0.5$, y $K = 40$ nodos de cuadratura, lo que da $C_1 = 6.036$ y $C_2 = 0.0739$. El algoritmo se ha aplicado con base $B = 5$, lo que proporciona intervalos de aproximación del tipo $[t_0, 25t_0]$.

Agradecimientos

El primer autor ha sido respaldado por la DGI-MCYT a través del proyecto MTM 2004-07194, cofinanciado con fondos FEDER. El segundo autor por DFG, SFB 382 y el tercero por DFG Research Center MATHEON "Mathematics for key technologies", Berlín.

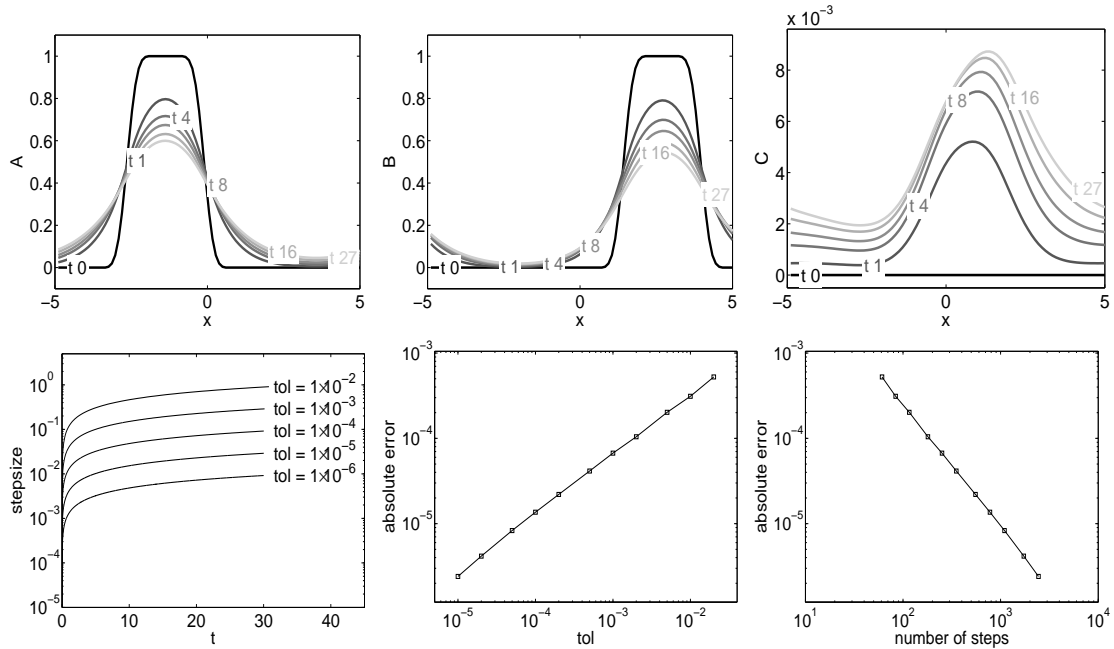


Figura 2: Arriba: Soluciones para las tres especies A , B , C , en tiempos diferentes. Abajo: Evolución del tamaño de paso, error vs tolerancias y error vs número de pasos.

Referencias

- [1] K. ADOLFSSON, M. ENELUND, AND S. LARSSON, *Adaptive discretization of fractional order viscoelasticity using sparse time history*, Computer Meth. Appl. Mech. Eng., 193 (2004), pp. 4567–4590.
- [2] X. CAO, K. BURRAGE, AND F. ABDULLAH, *A variable stepsize implementation for fractional differential equations*, Report, 2006.
- [3] R. HIPTMAIR AND A. SCHÄDLE, *Non-reflecting boundary conditions for Maxwell's equations*, Computing, 71 (2003), pp. 265–292.
- [4] M. LÓPEZ-FERNÁNDEZ, C. LUBICH, AND A. SCHÄDLE, *Adaptive, fast and oblivious convolution in evolution equations with memory*, Preprint 2006.
- [5] M. LÓPEZ-FERNÁNDEZ AND C. PALENCIA, *On the numerical inversion of the Laplace transform of certain holomorphic mappings*, Appl. Numer. Math., 51 (2004), pp. 289–303.
- [6] M. LÓPEZ-FERNÁNDEZ, C. PALENCIA, AND A. SCHÄDLE, *A spectral order method for inverting sectorial Laplace transforms*, SIAM J. Numer. Anal., 44 (2006), pp. 1332–1350.
- [7] C. LUBICH AND A. SCHÄDLE, *Fast convolution for nonreflecting boundary conditions*, SIAM J. Sci. Comput., 24 (2002), pp. 161–182.
- [8] A. SCHÄDLE, M. LÓPEZ-FERNÁNDEZ, AND C. LUBICH, *Fast and oblivious convolution quadrature*, SIAM J. Sci. Comput. 28 (2006), pp. 421–438.
- [9] F. STENGER, *Approximations via Whittaker's Cardinal Function*, J. Approx. Theory, 17 (1976), pp. 222–240.
- [10] F. STENGER, *Numerical methods based on Whittaker Cardinal, or sinc Functions*, SIAM Rev., 23 (1981), pp. 165–224.
- [11] S. B. YUSTE, L. ACEDO, K. LINDENBERG, *Reaction front in an $A+B \rightarrow C$ reaction-subdiffusion process*, Phys. Rev. E 69, (2004), pp. 036126.